USB Data Acquisition System

Hardware & Software Manual

Model UD128A8D

Documentation Number UD128A8D-4602

.

This product designed and manufactured in Ottawa, Illinois USA
of domestic and imported parts by

B&B Electronics

International Headquarters

B&B Electronics Mfg. Co. Inc.
707 Dayton Road -- P.O. Box 1040 -- Ottawa, IL 61350 USA
Phone (815) 433-5100 -- General Fax (815) 433-5105
Home Page: www.bb-elec.com
Orders e-mail: orders@bb-elec.com -- Fax (815) 433-5109
Technical Support e-mail: support@bb.elec.com -- Fax (815) 433-5104

European Headquarters

B&B Electronics Ltd.
Westlink Commercial Park, Oranmore, Co. Galway, Ireland
Phone +353 91 792444 -- Fax +353 91 792445
Home Page: www.bb-europe.com
Orders e-mail: orders@bb-europe.com
Technical Support e-mail: support@bb-europe.com

TABLE OF CONTENTS

B&B Electronics Mfg Co – 707 Dayton Rd - PO Box 1040 - Ottawa IL 61350 - Ph 815-433-5100 - Fax 815-433-5104
B&B Electronics Ltd – Westlink Comm. Pk. – Oranmore, Galway, Ireland – Ph 353-91-792444 – Fax 353-91-792445

Chapter 1: *GENERAL INFORMATION*

Introduction

The UD128A8D Data Acquisition System provides hardware and software to monitor Analog Voltages or 4-20mA current loop signals, to read on/off conditions, to set on/off outputs, and to output analog voltages for voltage controlled devices such as motor control circuits or variable power supplies. Minimal external circuitry is required for common I/O functions

Features

- Powerful Data Acquisition System in lightweight, compact and portable package
- USB connection provides Plug-N-Play support for multiple unit operation under Windows.
- USB 1.1 Full Speed Interface provides fast data transfer
- Power Supplied by USB power bus
- Multi-Function module supports A/D, D/A and Digital-I/O
- Eight 12-bit A/D Input Channels software selectable in pairs as single ended or differential A/D Input Channels
- High A/D input impedance minimizes source loading
- Four 10-bit D/A Output Channels
- 8 Digital I/O lines, software selectable as 8 Inputs, 8 Outputs or 4 of each.
- Built-in Digital Reference Voltage for +5V pull-ups
- Includes BB-DAC Application Software for controlling, displaying, measuring and saving real world signals in minutes.
- Visual, Graphical or Text Display in User defined units
- Log to File
- Includes UD Programmer's Library For Visual Basic programming

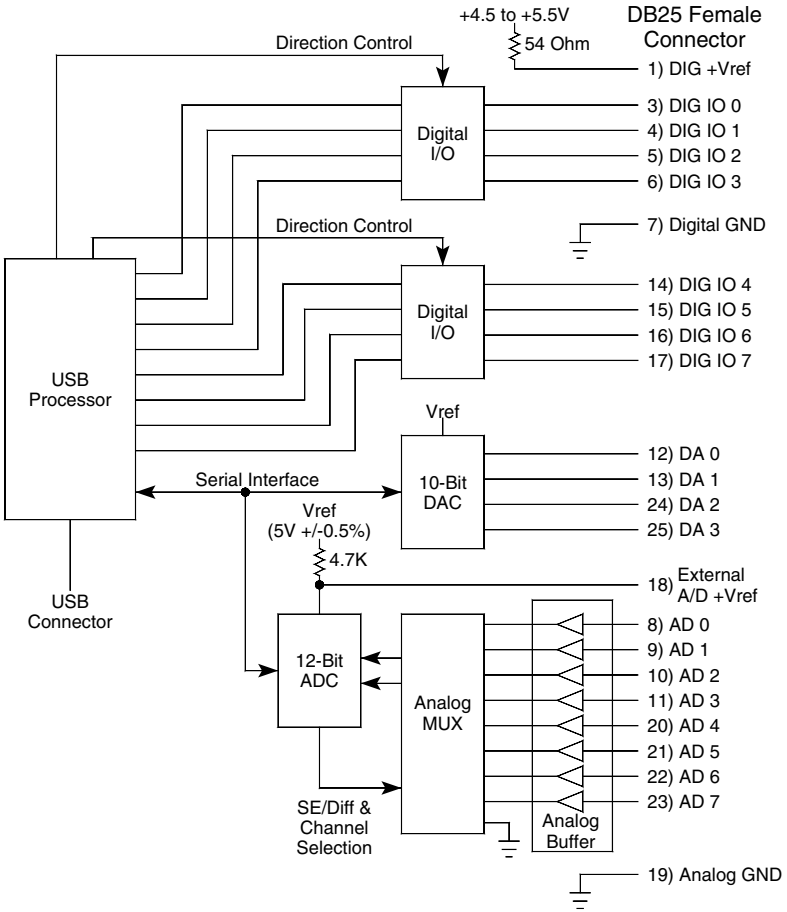


Fig. 1.1 - UD128A8D Block Diagram

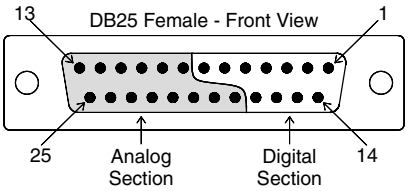


Fig. 1.2 – DB25F I/O Connector Pin Orientation

B&B Electronics Mfg Co – 707 Dayton Rd - PO Box 1040 - Ottawa IL 61350 - Ph 815-433-5100 - Fax 815-433-5104
B&B Electronics Ltd – Westlink Comm. Pk. – Oranmore, Galway, Ireland – Ph 353-91-792444 – Fax 353-91-792445

Description

UD128A8D Data Acquisition System Contents

The Hardware module provides 8 A/D analog inputs, selectable in pairs as single ended or differential inputs, 4 D/A analog outputs and 8 Digital inputs/outputs in two banks which can be software configured for either input or output. The module uses USB connections for power and fast communication with the host computer.

See Fig. 1.1 for the Module Block Diagram.

The Driver software provides Plug and Play compatibility with USB enabled versions (98/SE/ME/2000/XP) of Windows. The driver supports addressing multiple UD128A8D modules up to the limits of system resources.

The BB-DAC Application Software displays data in a graphical format. Values can be scaled and custom labeled and logged to a file for analysis in another application such as a spreadsheet

The UD Data Acquisition Library for Visual Basic provides functions to use in developing your own custom program with Visual Basic 6. Includes simple example code for using each function.

See the Quickstart section on the next page to get started.

Quickstart

1. Run Setup to Install the drivers and application software.
2. Make the Input/Output connections.

   See Chapter 2, Operational Check of Module.
3. Connect the module to the USB cable.
4. Start BB-DAC Application and load the Example file
5. Check the Analog Inputs and Outputs.
6. Check the Digital Inputs and/or Outputs.
7. Familiarize yourself with controls and options.

   See Chapter 4 for BB-DAC Design Controls and options.
8. Disconnect the USB connector from the module or computer.
9. Make the connections for what you want to read or control.
10. Make Safety Checks for hazardous voltages.
11. Connect USB cable to the module.
12. Start BB-DAC Application and load your custom design or the Example file.
13. If writing a custom application, use the recommended Operational Check connections to check/verify your program operation and perform calibration. Refer to Chapter 5.
14. Monitor/Control your devices and collect data.

Specifications

Digital Inputs/Outputs

Inputs/Outputs: 8 channels

- Group selectable as 8 Inputs, 8 Outputs or 4 of each.

Note: (a) Power-up clears and sets all digital channels to input.

(b) The last value output is held (latched) until changed.

(c) The input value read is most recent when read.

Input Ratings

- Vin High: 2.0V minimum
- Vin Low: 0.8V maximum
- Input leakage current: +/- 1.0μA max.

Output Ratings

Recommended Load Impedance: 209 ohms or greater

Minimum Load Impedance: 209 ohms

Maximum Sink/Source Current: 24mA per output

Vout High: 5V nominal, 3.76 V minimum at Iout = 24mA

Vout Low: 0V nominal, 0.44 V maximum at Iout = 24mA

Digital Reference Output Voltage

Line Name: DIG +Vref

+5VDC output for digital pull-up, +4.5V minimum

Output Impedance: 54 ohms

Analog Inputs

Inputs: 8 channels

- Pair selectable as Single Ended or Differential inputs
- Differential input pair channels: 0-1, 2-3, 4-5, 6-7
- Max Voltage on Differential input(-) pin: 1.25V

Resolution: 12-bit A/D input: 4096 Steps (values 0 to 4095)

Input Impedance: >10 Tera ohms (x10 to 12^{th} power)

Input Range: 0 to 5 volts (default internal ref. w/o external scaling)

Linear Range using built-in 5 volt reference: +0.1 to +5.0 V

A/D Reference: Internal or External

Internal Reference: +5V ± 0.5%

External Reference: Shunt type, minimum value 1.0V

Sampling Types: Hardware Timed or Software Polled

Hardware Timed sampling started/stopped by software command

Minimum Time between samples (Hardware Timed Mode): 62.5µs per channel (16K samples per second for 1 channel, 2k/sec for all 8 channels)

Maximum Sampling Time error: ± 0.5us

Overvoltage Protection: -10 to +15 V indefinitely

Maximum current draw at maximum overvoltage: 11mA

Max DC offset: ± 3 LSB and ±0.75mV (Total for 5V internal reference is 4.41mV)

Max Gain error: ± 4 LSb and ±Vref error (Total for 5V internal reference is ±0.1% + 0.5% = 0.6%)

Integral Linearity error (offset from best linear fit. Can't be calibrated out.): ± 2 LSB (± 2.44 mV with internal 5V reference)

Total possible calibrated FSE: ±0.05%

Total possible uncalibrated error using 5V internal reference: (Reading * .006) + .00441 + .00244 (This is 0.03685V at 5V full scale.)

Total possible uncalibrated FSE: ± 0.737%

Note:

Analog input values are the most recently read value and may include the desired signal plus any noise and DC offset component.

The accuracy of any measurement equipment depends on the signal integrity entering it. To realize the full potential of the UD128A8D, pay careful attention to signal grounding, use input line shielding and twisted pair wires to the inputs. Analog filtering for out-of-band signals and/or digital filtering using oversampling may be required to meet end requirements.

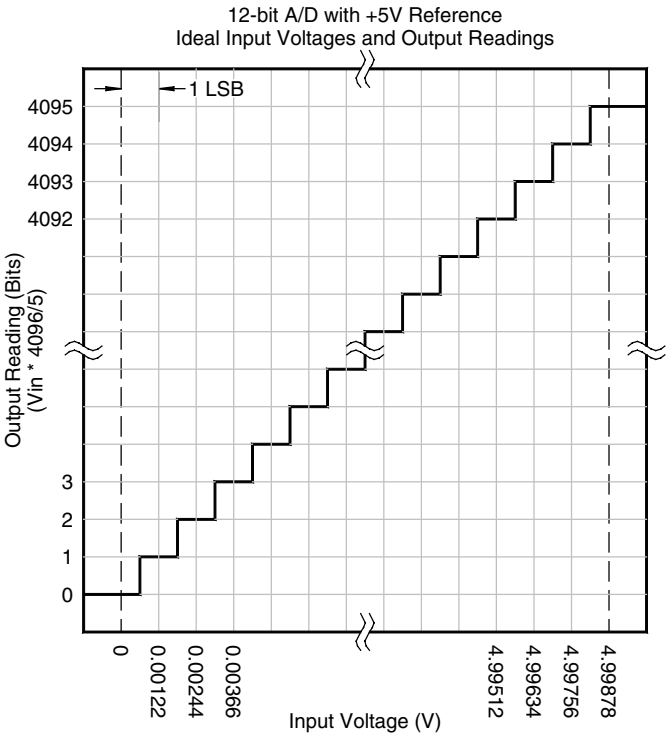


Fig. 1.3. 12-Bit A/D Digital Steps to Input Voltages

Analog Outputs

Outputs: 4 Channels

- Single Ended, Voltage output

Notes: (a) Power-up clears and sets output to high impedance state.

(b) The last value output is held (latched) until changed.

(c) There is no hardware readback function for last setting

Resolution: 10-bit A/D Output: 1024 Steps (values 0 to 1023)

Linear Range: 0 to 4.995 (5V – 1 LSB)

Maximum load Capacitance: 100pF

Output Impedance: ≈ 5 ohms maximum

Recommended Load Impedance: 10K ohms, resistive

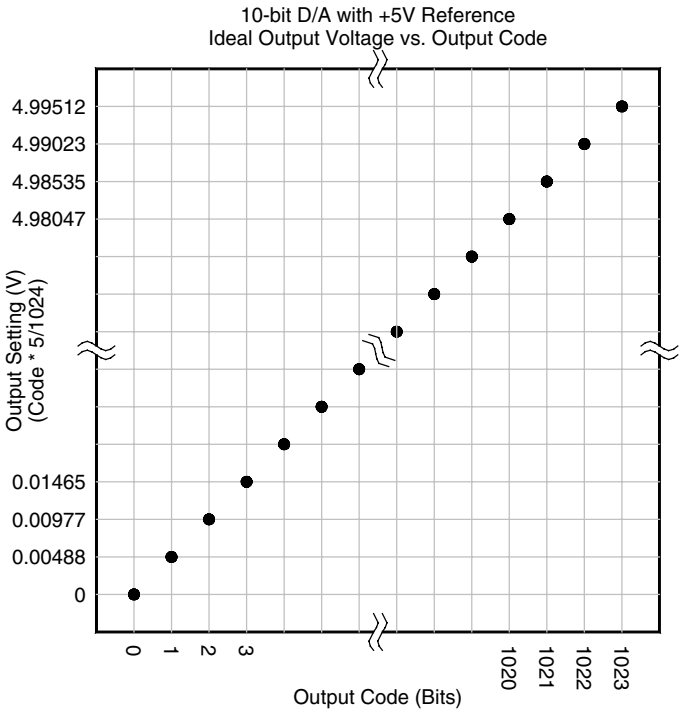


Fig. 1.4. 10-Bit D/A Voltages to Steps

Maximum Load current: 2.5mA per channel (2k Ω resistive load minimum for 5V output)

Integral Non-linearity error: ± 4.88 mV

Zero Scale Error (Offset error at 0 scale): ± 12 mV

Max Gain Error: ± 2.1%

Total Maximum Full Scale Error: ± 122 mV

Power Source

Power Supplied by USB power connections

Current Draw: Full power device. May draw as much as 500mA after configured.

Operation through a USB hub requires that the hub provide power for itself and downstream devices

General Specifications

Interface

USB 1.1 Full Speed (compatible with 2.0 controllers and hubs)

Connectors

USB Device: USB Type B, female

I/O: Single 25 pin female D-sub (DB25F) connector

Dimensions

3.8 x 2.4 x 1.1 in (9.7 x 6.1 x 2.8 cm) (without connections)

Environmental

Operating Temperature Range: 0º to 50º C

Operating Humidity: 0 to 95%, Non-condensing

Operating Systems Supported

Windows 98, SE/ME, 2000 and XP

Supplied Accessories

Software:
CD-ROM Disc for Windows 98/2000/XP
(includes: Drivers, BB-DAC software and VB Control Library)

Manual:
Instruction Manual (paper copy of this booklet)

Optional Accessories

1. USB Type A Male to Type B Male cable (USBAMBM-2M)
2. 25 Pin Terminal Block Adaptor (DTB25)
3. DB25 Male connector (DB25P)
4. Channel Relay Module (SDDRB4)
5. Add-in 2 Port USB Host Adapter (USBHA) for PCI

Notes on Operating Characteristics

General

- Each UD128A8D has a unique serial number. When it is installed, it will register itself and be given a name DAQx where "x" is a sequential number starting at 1 and increasing for each new unit installed on that computer. Software can be written to address the first one plugged in or by the "DAQx" name. The method is similar to addressing COM ports, where the first one installed gets called COM1, the next one COM2, etc. The uninstall should take out all instances, so the next one plugged in after an uninstall, and new install, will be named DAQ1.
- The UD128A8D enumerates as a full powered device, which means it will request as much as 500mA from the USB bus. For full operation it must be connected into the host or a externally powered hub which has a power supply.
- If you want to trigger on a digital input or analog input value your software must read and act on the value. There are no hardware triggers built in.

Analog Inputs

- A zero voltage input (Input grounded) produces a reading of 5 or 6 LSbits (6 or 7mV). The input buffer does not allow the input to go any lower than this.
- All unused inputs should be grounded.
- Maximum sampling rates will be best when this is the only device on the USB. It may be necessary to add a USB controller rather than a hub to maintain throughput.
- A capacitor of 0.1 to 1uF works well as a low pass filter for slow signals that may be noisy or some distance from the UD128A8D.
- There is no isolation in the device. Ground noise and grounding differences should be kept to a minimum.
- Timed sampling takes place in sequential order as close together as possible. The time between samples for each channel should remain constant. The analog channels go through a multiplexer which selects each channel.
- The Analog Input reference is set to 5V, but can be overdriven by an external reference. (See Fig. 2.19)

The analog inputs are capable of being configured as differential pairs, but have some rules.

1. The pairs are locked with the even numbered inputs being the positive input and the odd numbered inputs being the negative (AD0 & AD1, AD2 & AD3, etc.)
2. The UD128A8D does not produce a negative output if the plus input goes below the minus. The plus input always has to be higher.
3. There is no common mode voltage tolerance. Both inputs must be in the 0-5V range and should still have a ground connection.
4. The full output range is still set by the reference. If you want to measure a differential signal that has less span than 0-5V, you must change the reference.
5. The minus input cannot go above 1.25V.

- The differential input is mainly for getting rid of ground noise.
- Digital and Analog grounds are tied together at one point on the PCB and should not be used interchangeably to avoid noise problems.
- When using the analog inputs to measure a 0-20mA signal with a resistor to ground, it has to be the last element in the loop, returning current to the supply. If more than one measuring device is used, the UD128A8D must be last or the loop must be isolated from ground. (See Fig. 2.16)
- To obtain the best accuracy of the UD128A8D using the built-in 5V reference, you must use a 249 Ohm resistor as the sense resistor for 0-20 or 4-20mA measurement. (See Fig. 2.16) The voltage for 20mA current is 4.98 Volts or 4080 counts. The voltage supply for the current loop must be more than 5V plus all other voltage drops in the loop.
- The digital count for 4-20mA is 204 counts per mA. All the 816 counts under 4mA (0.996 volts) are below scale, so useable range is from 816 to 4080, 3264 counts over the 16mA range. This is the best possible from 4096 steps without using additional external circuitry.

Analog Outputs

- Analog output commands get second priority to Analog inputs during timed sampling. If there is not enough time to set an analog output, an error will be reported from the hardware. If there is time to set an analog output, it will be set between reads of the A/D. All commands have the same priority when timed sampling is not taking place. Digital I/O can be done during Analog Sampling, but may be slower depending on the sampling rate
- The Analog Output reference is fixed at 5V. Factory hardware modifications are required to set a lower reference.

Digital I/O

- The digital reference is intended primarily for pull-ups on the digital lines. Current output is limited by a series resistor, so little external circuitry can be powered. Because it is connects to all the digital circuitry, it's a bad idea to use it for any analog input signal conditioning.
- Digital channels have to be polled. Polling speed depends on the PC speed, OS, and what other apps are running. Our tests here indicate a 4-8 millisecond polling interval. If you need to catch a shorter period signal, you must add external latch circuitry.

Calibration

- Input or outputs should not be calibrated exactly at the rails (0V, 5V) one diode drop below the rails is easy to achieve and provides a fairly stable voltage.
- Input and output calibration values can most easily be determined by connecting one of the D/A outputs into one of the A/D inputs to compare input & output readings with your standard.
- Wait for calibration voltage to stabilize.
- Make sure calibration is at operating temperature.
- Make sure to ground any unused inputs.
- Make sure to read calibration voltage with UD128A8D connected.
- The more accurate the calibration test equipment the better.

Chapter 2: *CONNECTIONS AND EXAMPLES*

Cautions

Avoid damage to your Data Acquisition Module inputs/outputs and computer and other equipment. *This is very important in dry climates and during the Winter months!*

CAUTION: ELECTROSTATIC SENSITIVE DEVICE

Use ESD precautions for safe handling.

Before removing the module from the anti-static protective packaging

- Discharge any static electricity buildup on your body by touching a large grounded metal surface or the metal chassis on equipment connected to earth ground by a 3-wire power cord.
- Avoid touching the connectors or I/O port connection wires until any static on your body has been discharged.
- Unplug the USB Cable before making connections to the I/O plug from your controlled or monitored devices.

Safety Checks

- After making connections to your I/O connector plug, make sure no hazardous voltages exist between the I/O Connection lines from your devices to the ground pins (A-GND and D-GND) by checking for AC and DC voltages with a Voltmeter or DVM. Then connect your I/O plug to the I/O Connector on the module.
- Check for AC and DC voltages between the A-GND or D-GND and the computer ground/chassis before connecting the USB cable to the module and computer.

Protect Your Module

The module can withstand minor overvoltage, it is protected up to +15V or –10V. Connections which exceed the ratings or which cause excessive input or output current may damage the module.

1. Don't short or connect the digital or analog outputs to ground or to a voltage source without current limiting resistance.
2. If your load has less resistance than the specified ratings, excessive current will damage the driver circuits. Use current limiting resistors in series with the output lines.

Operational Check - Connections

A Digital voltmeter or DC coupled oscilloscope is recommended for checking signal levels and noise.

Use a Terminal Blocks to DB25 connector for preliminary connections or make your connections using the terminals on your prototype interface board (prototyping board not included).

Steps *- Refer to the I/O Pinouts (facing page)*

1. Connect D/A output channels 0-3 to 4 of the A/D input channels (Connections can verify D/A outputs and A/D inputs)

2. Connect a variable voltage source to the remaining A/D channels. (as below) or connect to ground or +5V.

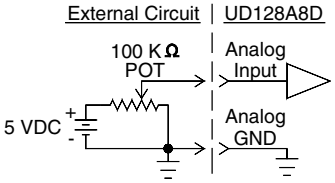


Fig. 2.1 - Analog Input using potentiometer for sensing position

3. Connect each digital output to a 2.2K to 3.2K resistor and LED.

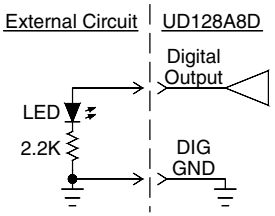


Fig. 2.2 - Digital Output Active High LED On

3. After starting the BB-Dac Application and defining outputs, test the digital inputs:

Connect each digital input to a 2.2K ohm resistor and then to ground or D-REF+5 (see Figure 2.3 or 2.4).

When done, disconnect test connections and connect your devices.

Common Connections – Example Figures are shown later in this chapter as a guide for commonly used inputs and outputs.

I/O Connector Pinouts

Description	Direction	Name	DB25F Pin #
+5V Pull-up Digital Vref	Output	DIG +Vref	1
Reserved	----	----	2
Digital I/O Ch0 – Group1	In/Out	DIG_IO_0	3
Digital I/O Ch1 – Group1	In/Out	DIG_IO_1	4
Digital I/O Ch2 – Group1	In/Out	DIG_IO_2	5
Digital I/O Ch3 – Group1	In/Out	DIG_IO_3	6
Digital Ground	---	DIG -GND	7
AD Input Ch0	Input	AD_0	8
AD Input Ch1	Input	AD_1	9
AD Input Ch2	Input	AD_2	10
AD Input Ch3	Input	AD_3	11
DA Output Ch0	Output	DA_0	12
DA Output Ch1	Output	DA_1	13
Digital I/O Ch4 – Group2	In/Out	DIG_IO_4	14
Digital I/O Ch5 – Group2	In/Out	DIG_IO_5	15
Digital I/O Ch6 – Group2	In/Out	DIG_IO_6	16
Digital I/O Ch7 – Group2	In/Out	DIG_IO_7	17
External AD +Vref	Output	ExtAD+Vref	18
Analog Ground	---	A-GND	19
AD Input Ch4	Input	AD_4	20
AD Input Ch5	Input	AD_5	21
AD Input Ch6	Input	AD_6	22
AD Input Ch7	Input	AD_7	23
DA Out Ch2	Output	DA_2	24
DA Out Ch3	Output	DA_3	25

Note: Digital I/O Group #1 or #2 is selectable for inputs or outputs.

Setting For Optional SDDRB4 4-Relay Buffer Board

UD128A8D Signal Name	DB25 Pin#	JP1:Position 12-15	Relay Channel#
DIG_IO_4	14	15	D
DIG_IO_5	15	14	C
DIG_IO_6	16	13	B
DIG_IO_7	17	12	A

Note: Only one SDDRB4 can be set to match outputs

Digital Input Examples

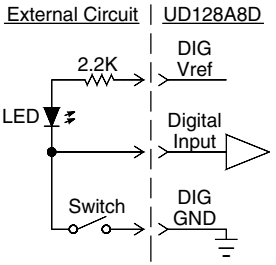


Fig. 2.3
Active Low with switch closure

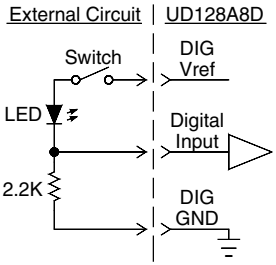


Fig. 2.4
Active High with switch closure

Note: LEDs may be omitted by increasing 2.2K to 4.7K ohm

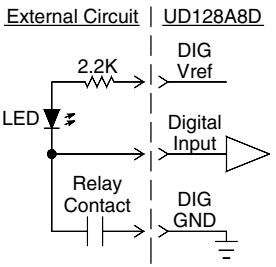


Fig. 2.5
Active Low with Relay contact closure

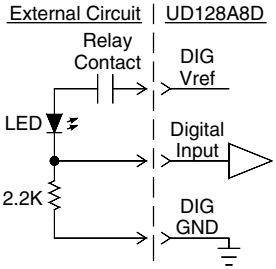


Fig. 2.6
Active High with Relay closure

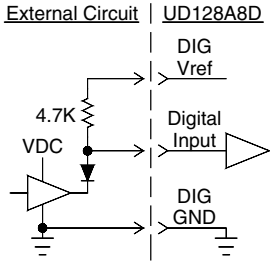


Fig. 2.7
Active Low pull-down

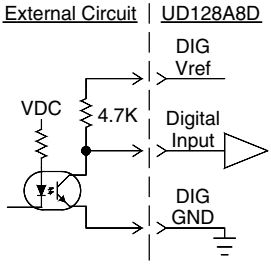
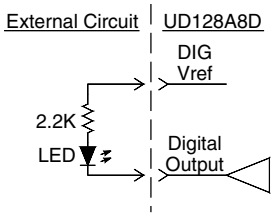


Fig. 2.8
Isolated AC/DC input Active Low

Digital Output Examples



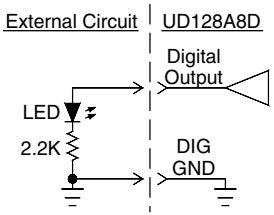
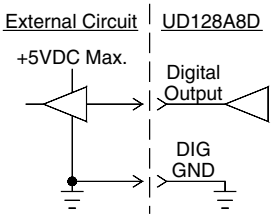


Fig. 2.9 Active Low LED On

Fig. 2.10 Active Hi LED On



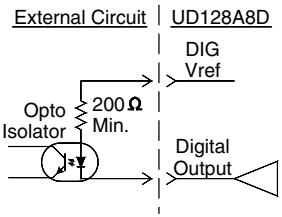
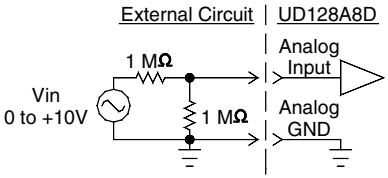


Fig. 2.11 Output to External Circuit/Buffer

Fig. 2.12 Output to Optical Isolator

Analog Input Examples



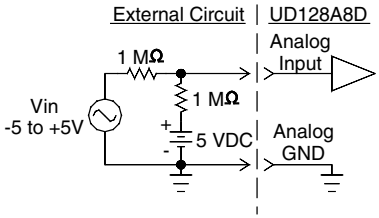


Fig. 2.13 Scaled 10 Volt Input Divided by 2

Fig. 2.14 Scaled 10 Volt Input Offset for +/-

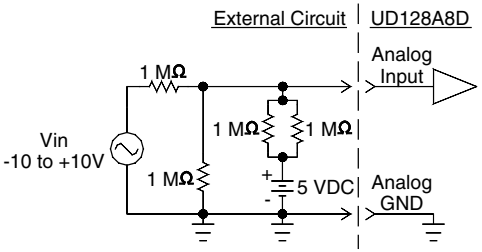


Fig. 2.15 Scaled 20 Volt Input Offset for +/-

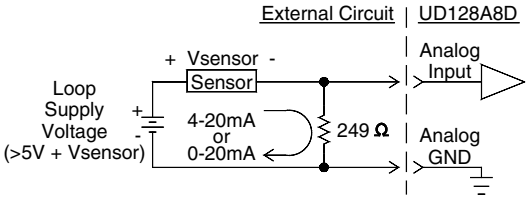


Fig. 2.16 Analog 4-20mA or 0-20mA Input

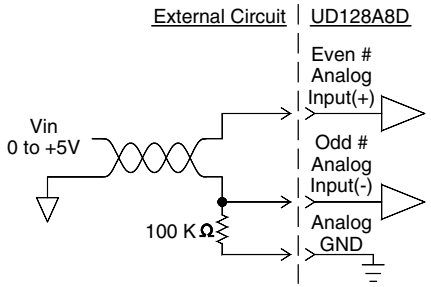


Fig. 2.17 Analog Differential Pair Inputs

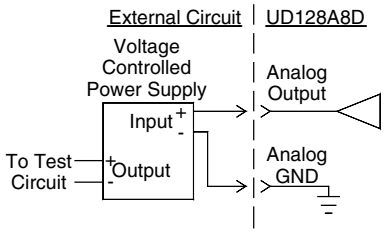


Fig. 2.18 Analog Output to Voltage Controlled PS

Scaling A/D Input Using Vref Adjustment

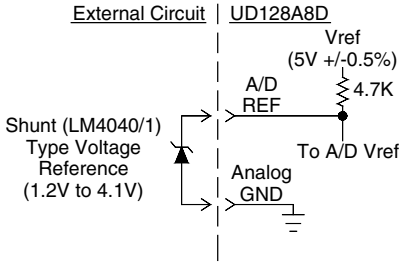


Fig. 2.19 – Using Zener or other Reference Diodes to Set F/S

Checking Input Calibration

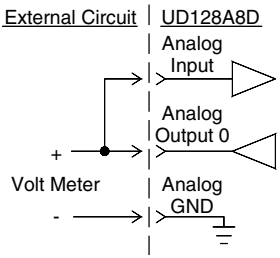
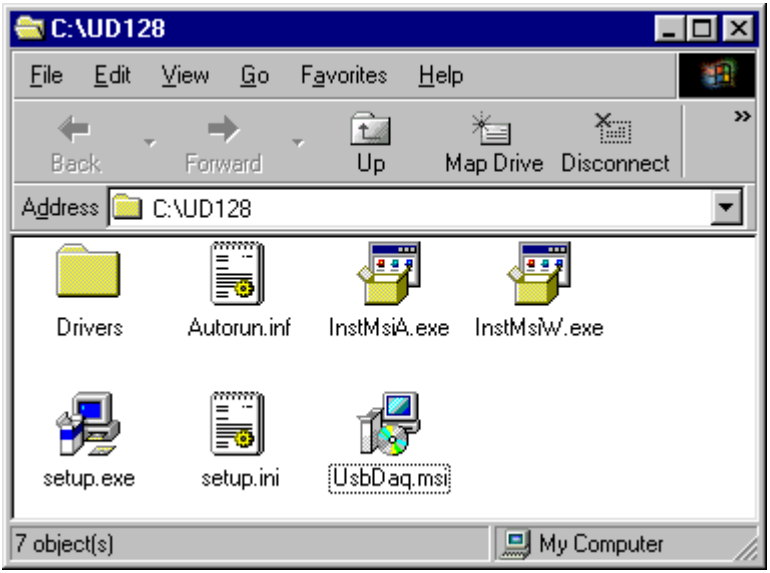


Fig. 2.20 – Checking Input Calibration Using the DA Output

Your Notes:

Do not start any software that wants to access the UD128A8D until after all installation is completed. If any such program is started, be sure to exit it before beginning.



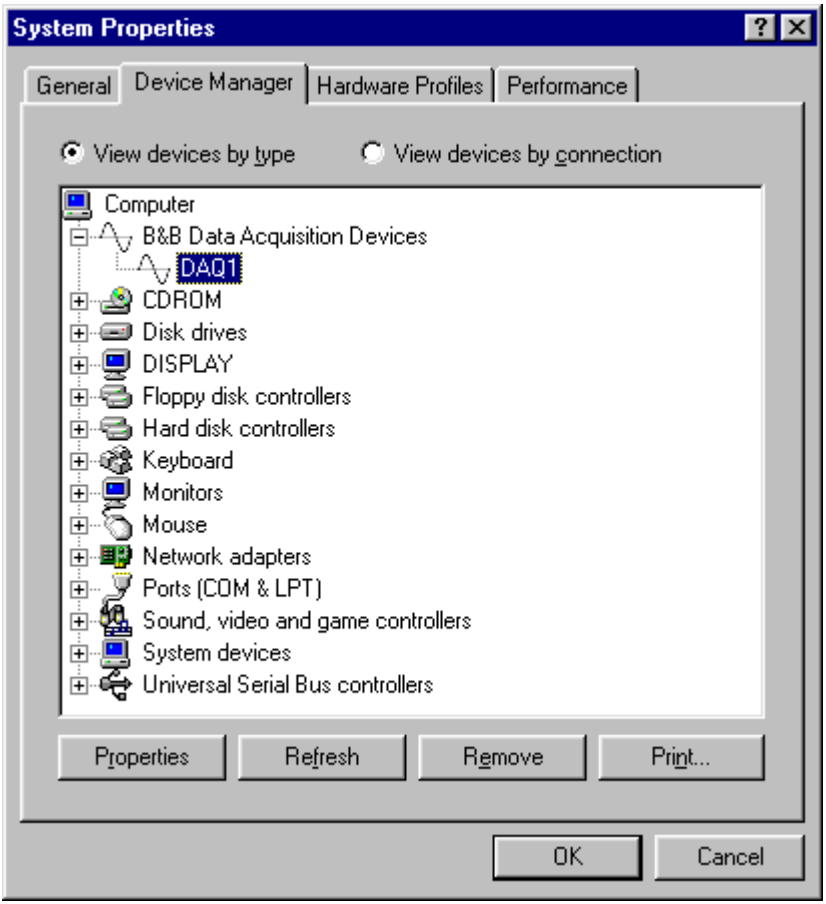
In the example above, the installation software was copied to a directory on the hard drive before installation. On the CD-ROM disc, the files should be in the root of Drive D: or whatever letter your CD drive is assigned.

Before connecting the UD128A8D module, insert the CD-ROM disc in the drive, locate Setup.exe and double-click it to start installation.

Click Next until software installation is completed, then click Finish.

Next, connect the USB cable to the module.

A new device is detected.



It should show up in the Device Manager as B&B Data Acquisition Devices, with DAQ1 as a child device.

The Data Acquisition Software is installed and accessed through Start, Programs, B&B Electronics, BB-Dac Application.

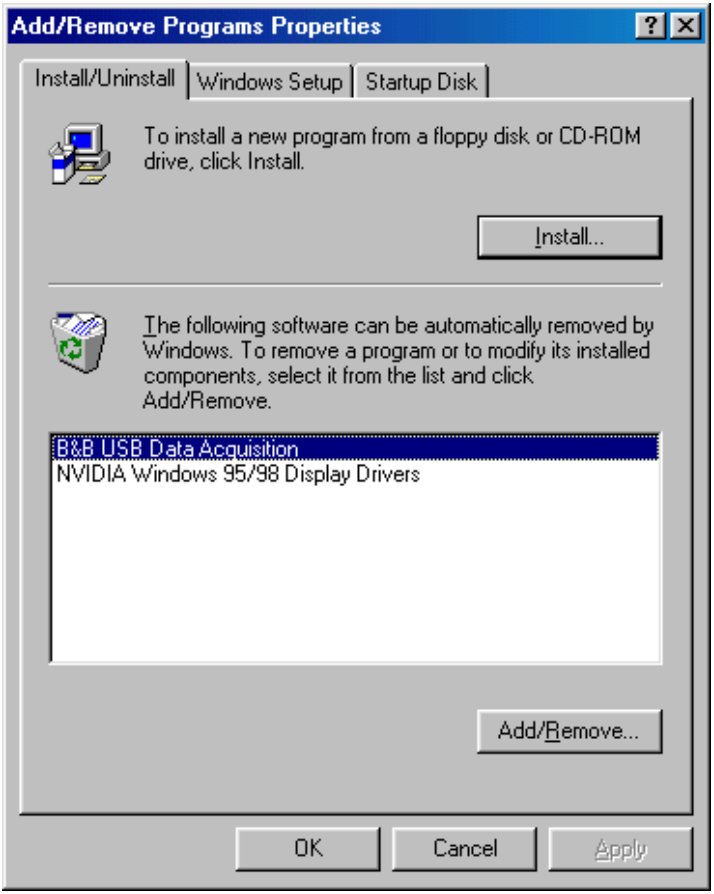
A BB-Dac directory is made on drive ***C:\My Documents\BB-Dac*** to access the BB-Dac Application Design example, and to save your new BB-Dac Designs. See the file, Sample1.daq

Example Programs written in Visual Basic are installed and accessed through Start, Programs, B&B Electronics, US128A8D, Example software.

The Visual Basic Programmer's Library is installed for use with your VB program. Refer to Chapter 5.

Removing Software

1. Open Control Panel, Select Add/Remove Programs.



2. Find B&B USB Data Acquisition in the Software Window. Select it.
3. Click the Add/Remove button.
4. Complete removal as needed.

Chapter 4 - *USING BB-DAC APPLICATION SOFTWARE*

The BB-DAC Application software is installed along with the UD128A8D drivers and VB Programmer's Library.

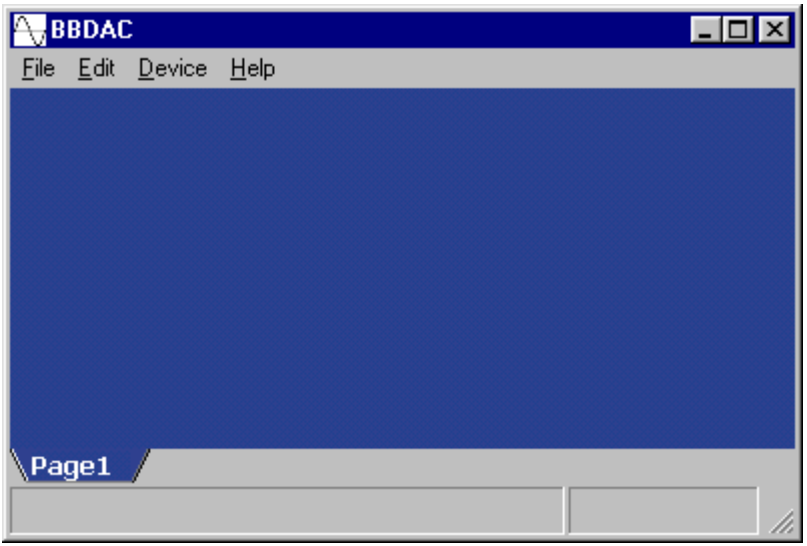
The BB-DAC Application consists of Control objects for indicating or setting On/Off (digital) conditions, and Control objects for indicating or setting analog values, such as voltage, current or relative levels.

Controls can be chosen for Control or Display by selecting one, then setting the attributes for display or output, and the I/O connection point.

BB-DAC has two modes, the Operate Mode or Design Mode. In the Operate Mode, previously defined designs can be loaded or the design currently loaded can be used. In the Design Mode, new application designs can be defined. Designs can be password protected from modification or change.

Start the BB-DAC Application

To access BB-DAC, use Start, Programs, B&B Electronics, BB-Dac Application.



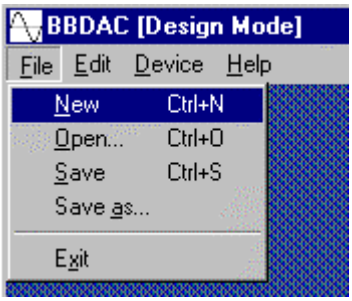
continued next page

File Menu

Use the **File**, Open menu to load your previously saved application or the Example application (located at C:\My Documents\BB-Dac).

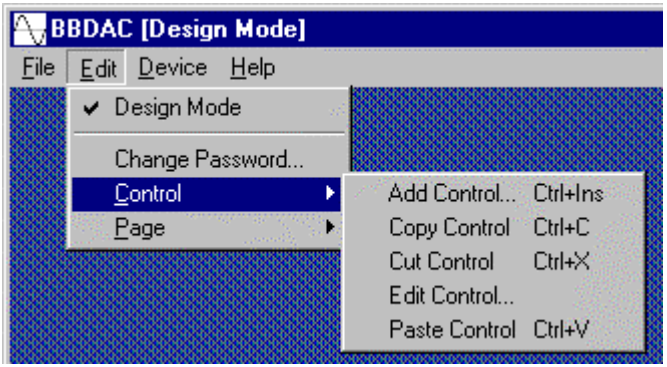
After loading an Application Design file, BB-DAC is set to the Operate Mode without input sampling enabled. *Start in next step.*

The **Device** Menu is used to Start Sampling or select Data Logging.



In the Operate Mode, only the **File** menu Open function is enabled. In the Design Mode, “New”, “Open”, “Save” and “Save as” are enabled.

Edit Menu



The **Edit** menu enables the Design or Operate Mode. When checked, the Design Mode is active, unchecked is Operate Mode.

If the Design Mode is selected and a Design is Password protected, you will be asked to enter the Password before the Design Mode becomes available.

After selecting Design Mode, you can Specify or Change the Password for your Design, Add Controls, Copy, Cut, Edit or Paste Controls.

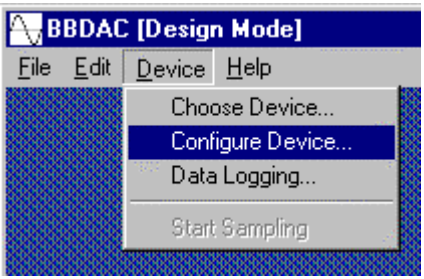
To begin your custom design of Controls, Display, and Functions, select **Edit**, **Design Mode**, then **File**, **New**. Next use the **Device** menu to chose your device and configure your I/O points.

When done defining your Controls, I/O Point connections, Display, and Layout, select **File**, Save or Save As, then deselect the Design Mode to Operate and Check your new design.

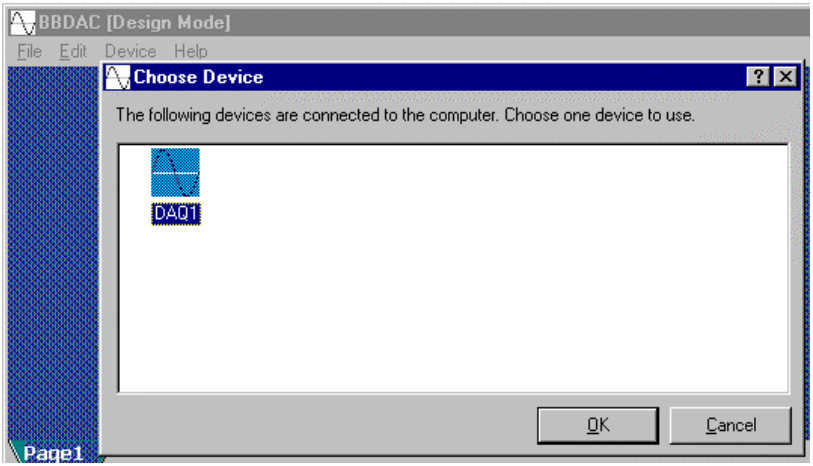
If you Password protect your Design to prevent it being changed by others in the Design Mode, be sure to write your Password down for each Design. Otherwise you will be unable to access it later to make changes. Make a backup copy of your design first!

Device Menu

After selecting the Design Mode, use the Device menu to Select your DAQ device. Next, Configure it for Input/Output functions, then for Sampling Interval.



Choose Device



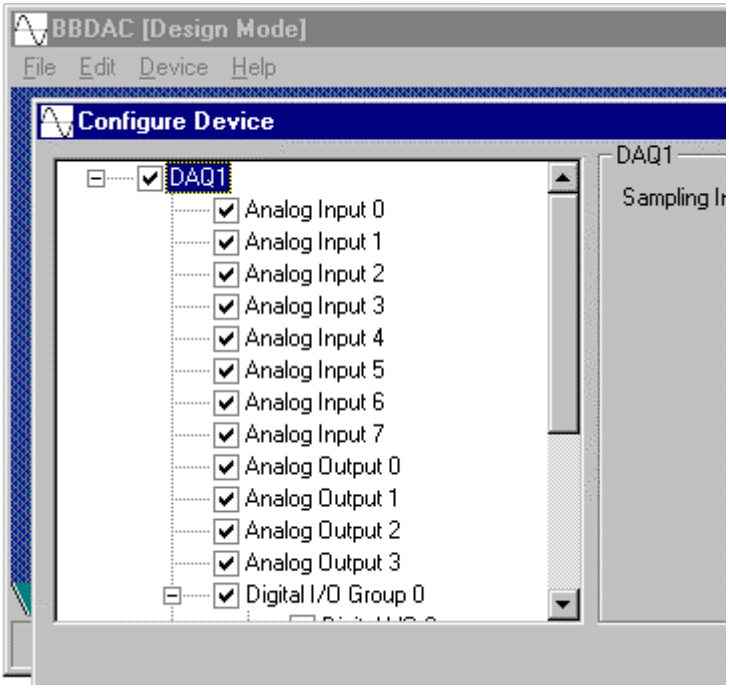
If a UD128A8D module is connected, you should have a single DAQ1 shown. If you have multiple modules installed, those connected should be shown, each with the DAQ number assigned in the original installation order.

Configure Device

After selecting the module, Configure the selectable I/O points as enabled or disabled, then for input or output as available.

In the UD128A8D, Digital I/O Group 0 and Group 1 default to Input, Analog inputs default to Input, Analog outputs default to Output.

Digital I/O Group 0 or Group 1 can be set independently for Output or Input. In the UD128A8D, Analog Inputs and Outputs are fixed to the standard mode, they can only be enabled or disabled.



Configure Digital I/O Group

To set Digital I/O Group 0 or Digital I/O Group 1 for input or output:

- Select the Digital I/O Group by clicking the Group box, then Set the Configuration for Digital Input or Digital Output mode.

Configure Analog Output Point

To enable a Analog output point:

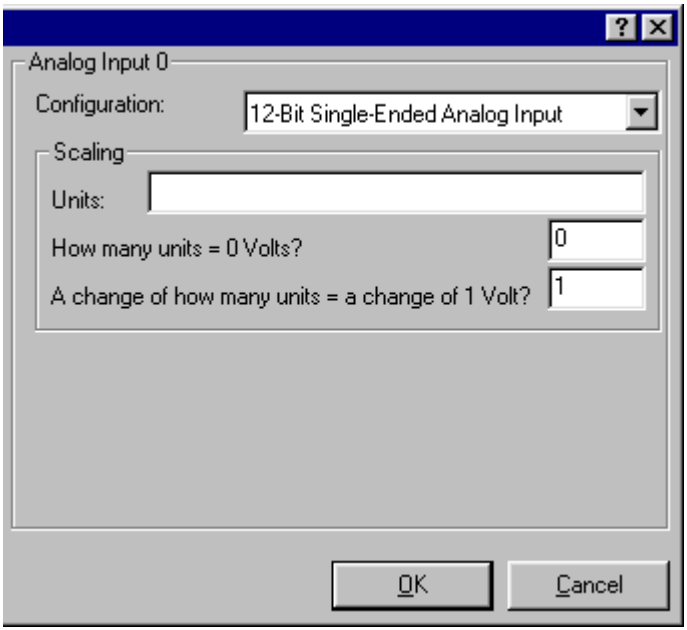
- Select it by Clicking the box to check it.

Configure Analog Input Point

To enable a Analog input point:

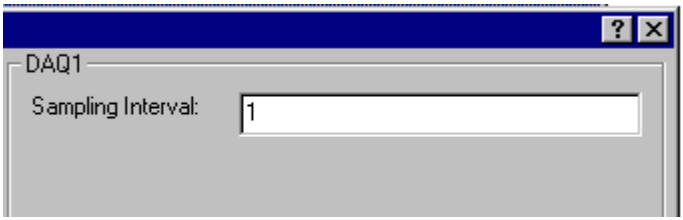
- Select it by Clicking the box to check it.
- Next set the Configuration for 12-Bit Single-Ended Analog Input or 12-Bit Differential Analog Input.
- Set the Units to Volts or another label.
- Set how many Units is equal to 0 Volts.
- Set how many Units equals a change of 1 Volt.

In the case of units with the standard +5 Volt reference and no external divider, a change of 1 volt =1 volt. If we use an external voltage divider to divide the voltage by one half, and our input voltage is 10 volts full scale, a change of 1 volt =2 volts. Enter 2 in the box. If we divide by 10, a change of 1 volt =10 volts. Using our standard +5 volt reference, a change in units will always be 1 or more.



During Sampling or Logging, only active points are sampled.

Sampling Interval



The Sampling Interval selects how often the I/O points are updated. The default is 1 second. The minimum interval is 0.0000625 (62.5usec) for sampling only one analog input, or 0.005 for all 8 analog inputs. To sample more points, the interval must be increased, because the analog input gets first priority and all other reads/writes must be interleaved between the analog reads. The Sampling Interval is used for display and for the log file.

Designing – Adding Controls

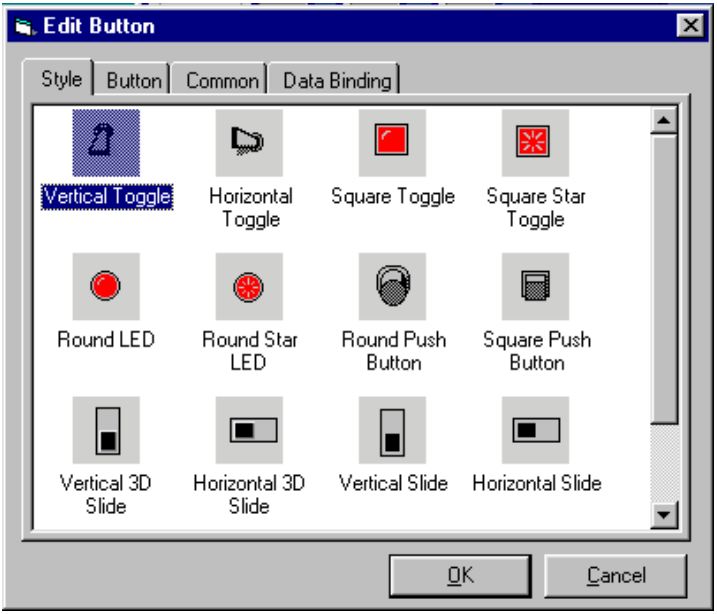
Use Edit, Control, Add Control, then select the Control.

Three classes of Control Styles are provided:

- Button Class – Digital Switches and On/Off Indicators
- Knob Class – Analog Output and Input Indicators
- Slide Class – Analog Output and Input Indicators

Each Class of Controls has different properties which can be defined by selecting the tabs as shown below for the Button class.

Button Class Control Style

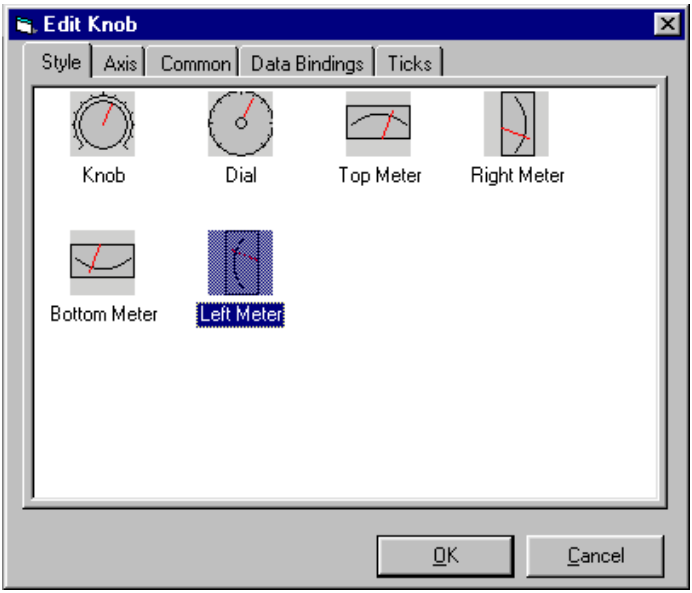


After you select a Control and Click Next, the program enters the Edit mode for the Control selected, and shows other controls in the same class with similar properties.

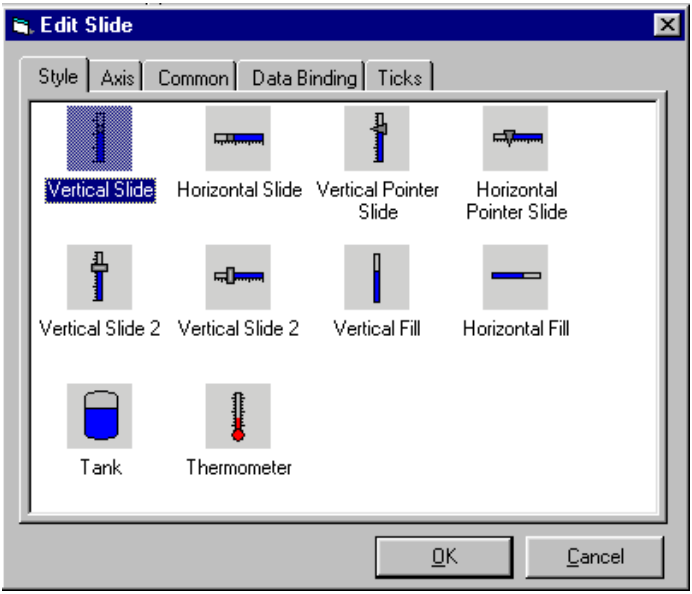
You can edit the Style initially selected to another control in the same Class if desired.

All styles have Data Binding to select which I/O point is to be used with the Control for Input or Output. See the other Classes which follow.

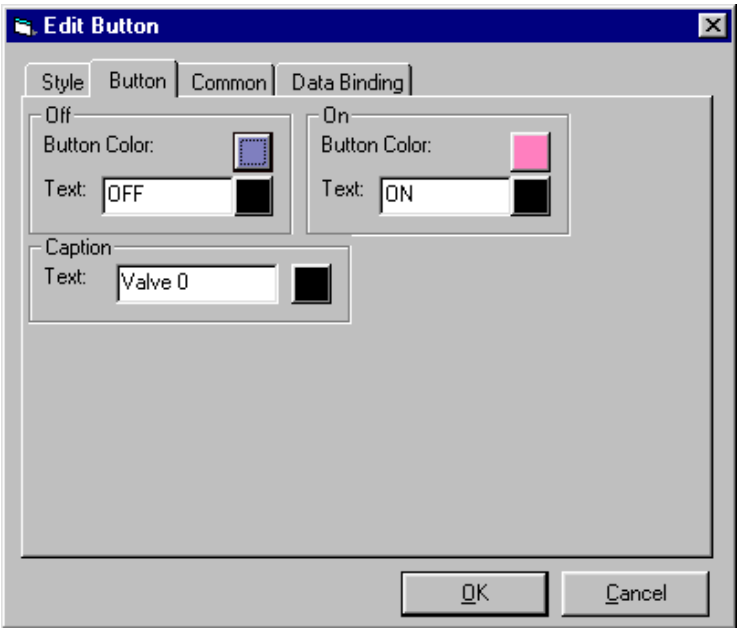
Knob Class Control Style



Slide Class Control Style



Button Class Properties



The Off Color and Text is shown when the Digital output or input is Low, the On Color and Text when High.

Enter the Text to be displayed when Off or On.

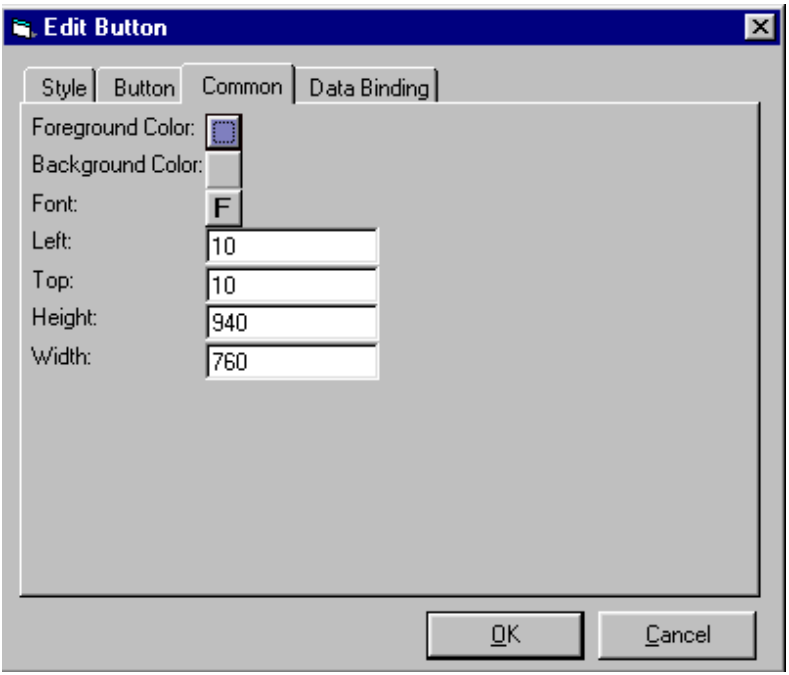
Click on the Button Color squares to Select the Off and On button colors, and on the Text Color square to Select the Text Color. Use light color for one indication, a darker color for the opposite.

Enter the Caption Text and Click the Color square to Select the Text Color. Use Dark captions on a light background, light captions on a dark background.

When done, Select Common to size and position the Control.

Continued next page

Common Properties



Next Select the Background color for the area outside the Off/On section (the background for the Caption Text), and the Foreground color. The Foreground color may already be set to the Off or On Color, if so, cannot be set independently.

The numbers for Left, Top, Height, Width establish the upper left position of the Control, and the Height and Width of the Control. These provide absolute positioning and sizing on the background.

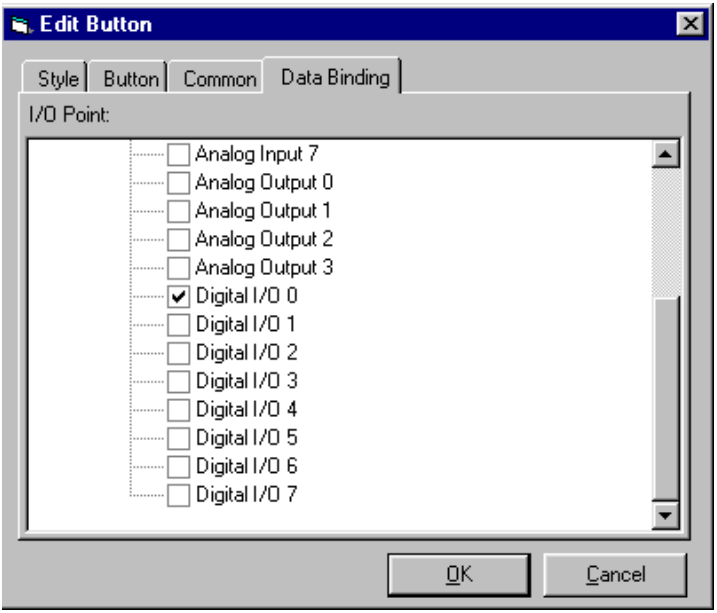
When done, Select Data Binding to chose the I/O Point the Control indicates or activates. Make sure the point selected is set to operate in the mode desired for the Control (see screen capture on next page.)

After Selecting Data Binding for the Control, select OK to finish with that control.

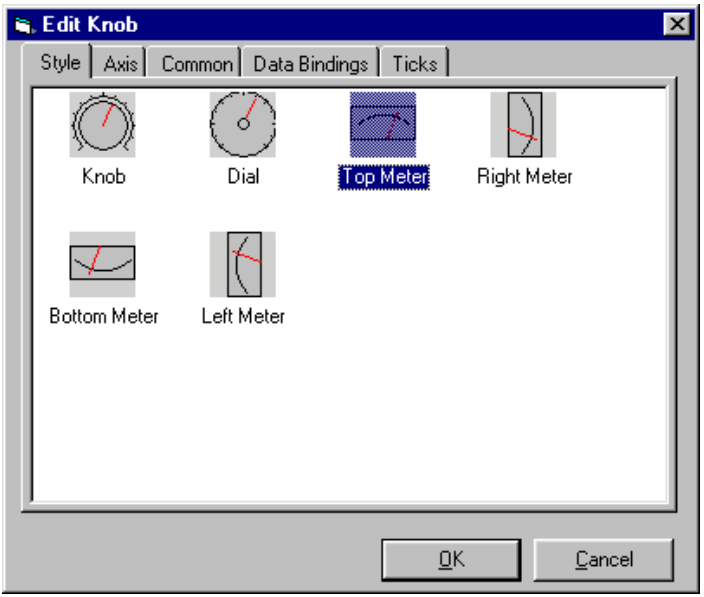
To Edit a Control already placed, select it, then right click, select Edit.

To Delete a Control, Select it, right click, select Delete.

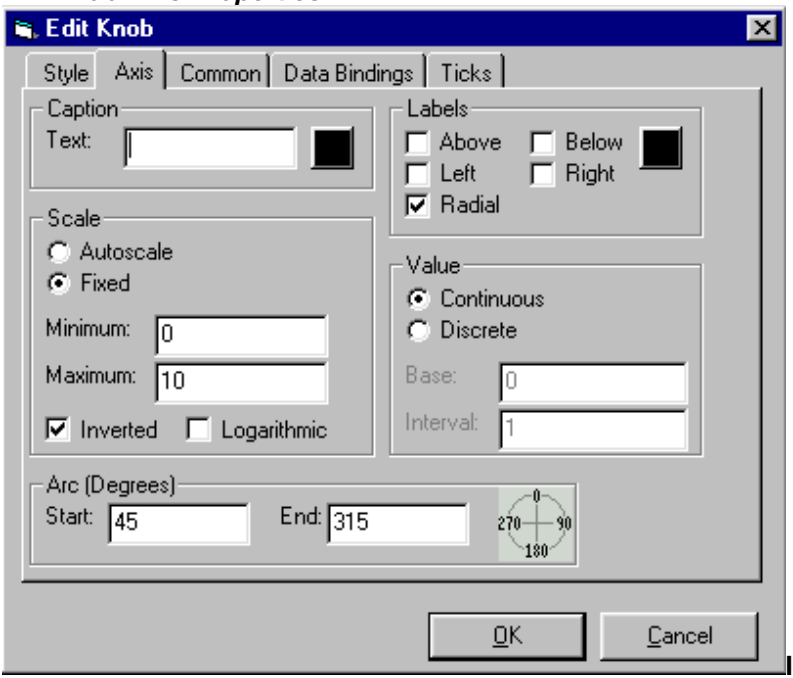
Data Binding I/O



Knob Class Styles



Knob Axis Properties



IS

The Axis Properties set the Control Caption, Caption Color, Scale Type, Minimum and Maximum, Arc Start and Arc End, Label type, Position, Color, and Value for Continuous or Discrete. These functions vary depending on the Control selected.

Axis Properties for Knob and Slider are nearly the same except that Arc is not available for Slider class Controls. (see next page)

The Min and Max set the End Values of the scale in Fixed mode.

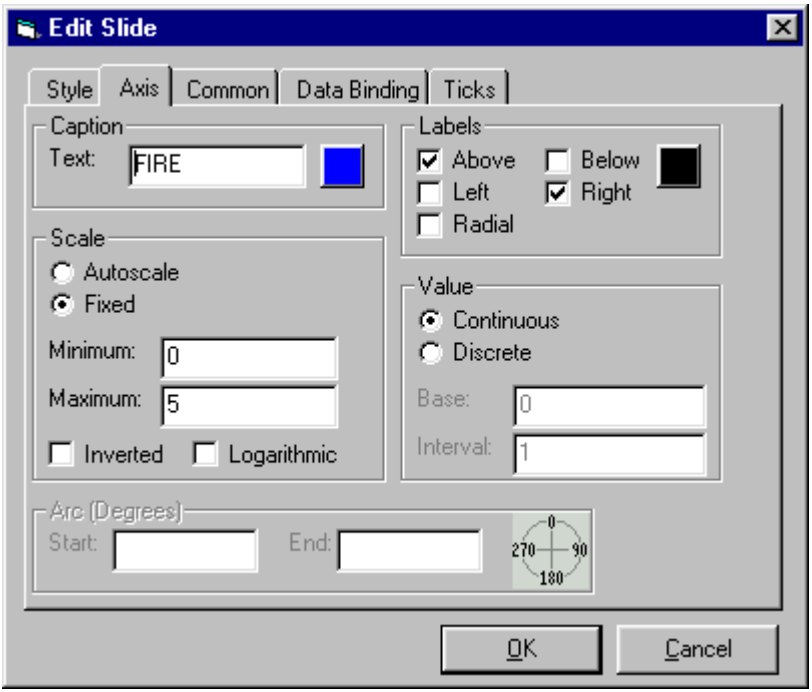
The Knob Common Properties sets the Control Colors, position and Size the same as in Button Common Properties.

The Data Bindings Properties selects the I/O point for Connection, the same as in Button Data Bindings. Remember, Configure Device set the Input Mode of operation and the voltage value for units.

The Ticks Properties Selects the major and minor Tick marks for the scale, tick positioning, colors.

Continued next page

Slide Axis Properties



The Axis Properties set the Control Caption, Caption Color, Scale Type, Minimum and Maximum, Label type, Position, Color, and Value for Continuous or Discrete. These functions vary depending on the Control selected.

Axis Properties for Knob and Slider are nearly the same except that Arc is not available for Slider class Controls. (see prior page)

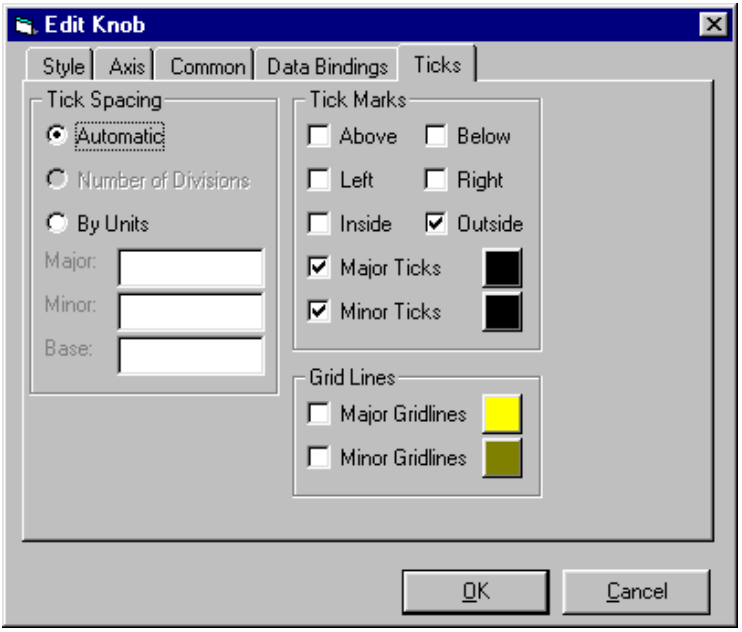
The Min and Max set the End Values of the scale in Fixed mode.

The Slide Common Properties sets the Control Colors, position and Size the same as in Button Common Properties.

The Data Binding Properties selects the I/O point for Connection, the same as in Button Data Bindings. Remember, Configure Device set the Input Mode of operation and the voltage value for units.

The Ticks Properties Selects the major and minor Tick marks for the scale, tick positioning, colors. (see next page)

Knob Tick Properties



The Ticks Properties Selects the major and minor Tick marks for the scale.

Say we want major ticks to mark at one volt, minor ticks in between at 0.2 or 0.25 volts. We previously set the Minimum Scale to 0 volts, Maximum to 5 volts. After we set Ticks, we will have a meter with markings and labels at 0, 1, 2, 3, 4, 5, with 5 or 4 ticks between each Volt mark. Some Tick settings don't apply to some types of controls so are ignored.

Select the tick types, units, spacing according to the size and type of control.

Ticks Properties for Knobs and Sliders use the same functions.

After Defining and positioning all your controls, save your file and de-select the Design Mode.

Check your functionally and display, adjust by editing until you achieve ease of use and clear meaning.

Chapter 5 - *UD PROGRAMMING WITH VISUAL BASIC*

Introduction

The UD128A8D Data Acquisition Library provides access to the hardware using the “B&B Electronics Data Acquisition Type Library 1.0” and Visual Basic Project References. Selection of the Library enables functions for reading or writing data to the UD128A8D. These functions are similar to those used with other devices.

The UD128A8D drivers make a collection of devices as found during startup or connection, this collection is accessed through the creation of global variables and objects. Provided functions include a means to determine how many devices are connected, to select which device is to be accessed, and to read or send data to any of the connected devices. An event is generated when a device is connected or disconnected. Your program must handle that event and take appropriate action. One program at a time can access the device collection using this version of the Library..

Devices connected can be addressed in the order in which they are found {Device(1), Device(2), etc.} during startup or by the DAQ name (“DAQ1”, “DAQ2”, etc.). If you plug in a new DAQ module, it is added to the collection as the next DAQ name (“DAQ3”) and will reserve and retain that name on that computer. If all of the devices are disconnected, and the last module is reconnected, it will be the first device found, Device(1), but could also be addressed by the DAQ name (“DAQ3”). Methods are provided for either type of module access.

The following sections covers the basics, then advanced access.

Description

The BBDaq Software Component provides an interface in Microsoft Visual Basic to the UD128A8D. It may be expanded later to cover other data acquisition devices with different functions.
Following is a description of the functions available using VB 6.0 for compatibility with Windows OS versions 98 through XP.

Library Software Component Organization

(1) BDaq.BBDaqEnum

A BBDaqEnum is a collection of connected UD devices. You can select a device from BBDaqEnum by name or reference. Connecting or disconnecting a device causes an event that can trigger the software to check connected devices and make appropriate changes.

(2) BBDaqDevice

A BBDaqDevice is a single UD device in the collection of devices.

(3) BBIOPointGroups

Each Device has groups of BBIOPoints.

A BBIOPointGroup is a set of I/O points that share a common set of traits or configuration options.

(4) BBIOPoints

An I/O point is a single input or output representing a real-world signal.

Enabling the Library for Use

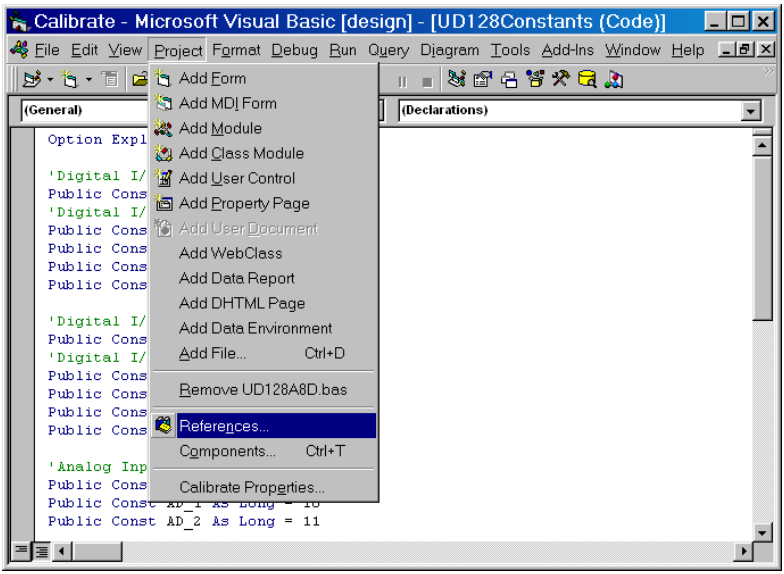
In order to use the Library, you must enable it in your program.

Install the UD128A8D software using Setup.

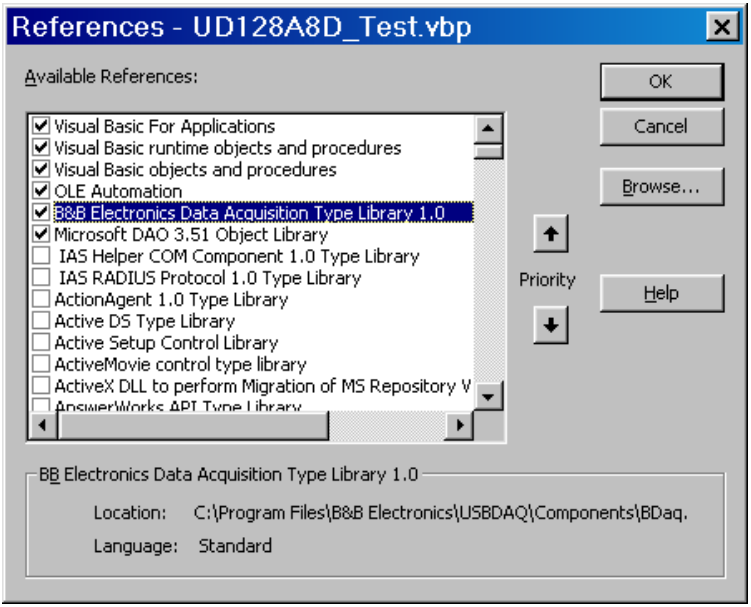
Start VB6.0

Create a new project

The library is installed in ***C:\Program Files\B&B Electronics\Data Acquisition\UD128A8D\Components .***



Select “References” from the “Project” menu item.



Click on the check box for "B&B Electronics Data Acquisition Type Library 1.0", located at: ***C:\Program Files\B&B Electronics\Data Acquisition\UD128A8D\Components).*** Then click OK.

Add the Global variables to the top of code for form1 before Form_Load. They become (Declarations) after entered.

To access a UD128A8D in VB:

In **DECLARATIONS** at the top of Form1:

1. Create a global variable to hold the enumerated devices.

```
Private WithEvents Connected As BDaq.BBDaqEnum
```

2. Create a global variable to hold each device that will be accessed.

```
Private WithEvents Device1 As BDaq.BBDaqDevice
```

3. Add additional device numbers you intend to access

```
Private WithEvents Device2 As BDaq.BBDaqDevice
```

CONSTANTS

4. Add the Constants for Point Groups and Configurations

' \ Add from the list or UD128A8D.Bas module supplied

```
'\***** Digital I/O Point Group 1 ****
Public Const DIG_GROUP_1 As Long = 1
```

```
'\** Digital I/O Points in Group 1
Public Const DIG_IO_0 As Long = 1
Public Const DIG_IO_1 As Long = 2
Public Const DIG_IO_2 As Long = 3
Public Const DIG_IO_3 As Long = 4

'\ **** Digital I/O Point Group 2 ****
Public Const DIG_GROUP_2 As Long = 2
'\** Digital I/O Points in Group 2
Public Const DIG_IO_4 As Long = 5
Public Const DIG_IO_5 As Long = 6
Public Const DIG_IO_6 As Long = 7
Public Const DIG_IO_7 As Long = 8

'\ **** Analog Input I/O Points ****
Public Const AD_0 As Long = 9
Public Const AD_1 As Long = 10
Public Const AD_2 As Long = 11
Public Const AD_3 As Long = 12
Public Const AD_4 As Long = 13
Public Const AD_5 As Long = 14
Public Const AD_6 As Long = 15
Public Const AD_7 As Long = 16

'\**** Analog Output I/O Points ****
Public Const DA_0 As Long = 17
Public Const DA_1 As Long = 18
Public Const DA_2 As Long = 19
Public Const DA_3 As Long = 20
```

I/O Point Group Configurations

```
'\**Digital I/O Point Group possible
configurations
Public Const DIGITAL_INPUT As Long = &H20000003
Public Const DIGITAL_OUTPUT As Long = &H20000004
Public Const DO_NORMAL As Long = &H80000004


'\**Analog Input Endpoint possible configurations
Public Const AD_12BIT_SE As Long = &H40000002
Public Const AD_12BIT_DIFF As Long = &H40000003

'\**Analog Output Endpoint possible configurations
Public Const DA_10BIT As Long = &HC0000005

' \ ** Flag for Invalid Data in Timed Samples
Public Const BDAQ_INVALID_DATA_VALUE As Long _
 = &H80000000
```

5. Create an instance of BBDaqEnum and set it to your variable.

```
Set Connected = CreateObject("BDaq.BBDaqEnum")
```

‘\ Connected is your variable - This should be done only once in

‘\ the program, probably when the startup form loads. This

‘\ enumerator must exist while any device is open.

Finding UD Devices

6. Next see if any UD devices are available

```
If Connected.Devices.Count > 0 then
```

‘\ 7. Set your device variables equal to each connected device.

‘\ By the order they were connected:

```
Set Device1 = Connected.Devices(1)
```

‘\ Or By Name: - The named device must be attached

```
Set Device1 = Connected.Devices(“DAQ1”)
```

```
 '\ 8. If we have one, Open it.
   Device1.Open
ELSE
 '\ 9. if none, cleanup and end program
 Set Connected = Nothing
 End
END IF
```

10. Before Running, make an a Exit Routine to cleanup & End

```
Public Sub CleanExit()
```

‘ \ VB’s End does not close or clean up – you must!

Clean-up – Closing - Exit

‘ \ Cleanup before End

```
   On Error Resume Next
   Device1.Close
```

' \ To remove the variable holding a device,

‘ \ Set it to Nothing.

```
   Set Device1 = Nothing
```

' \ To remove the list of connected devices,

‘ \ set it to Nothing.

```
  Set Connected = Nothing
```

' \ Any alias I/O point variables created should

‘ \ also be set to Nothing

' \ Set ad0 = Nothing

```
End
End Sub
```

11. Next Set the Configurations for Inputs and Outputs

Digital I/O Configuration

```
Device1.IoPointGroups(DIG_GROUP_1).Configuration _
= DIGITAL_INPUT
```

‘ \ Sets Digital I/O 0-3 to Inputs

‘ \ The space underscore (_) is the VB line continuation char

```
Device1.IoPointGroups(DIG_GROUP_2).Configuration _
= DIGITAL_OUTPUT
```

‘ \ sets Digital I/O 4-7 to Outputs

Note: The UD128A8D has only 1 type of digital I/O (No latches or counters), so no other configurations are necessary. Future units may require that each I/O Point be configured.

Analog Input Configuration

```
Device1.IoPoints(AD_0).Configuration = AD_12BIT_SE
```

‘\ Set AD0 to a single-ended input.

```
Device1.IoPoints(AD_0).Configuration _        =
AD_12BIT_DIFF
```

‘\ Set AD0 and AD1 differential input + and - respectively.

```
For X = 0 To 7
    ‘\ Set all inputs to Single Ended inputs
  Device1.IoPoints(AD_0 + X).Configuration _
  = AD_12BIT_SE
Next X
```

Analog Output Configuration

```
Device1.IoPoints(DA_0).Configuration = DA_10BIT
For X = 0 To 3
      ‘\ Enable all D/A Outputs
  Device1.IoPoints(DA_0 + X).Configuration _
  = DA_10BIT
Next X
```

Reading Analog I/O Points

Reading or Setting an I/O point uses the Value property.

The A/D channel Value property is Read Only and returns the reading in counts.

```
Dim Reading as Long
Reading = Device1.IOPoints(AD_0).Value
```

‘\ Reading is a count from 0 to 4095. Value is +Vref/4096

With a 12-bit A/D there are 4096 possible values compared to the reference voltage. The value of each step depends on the reference voltage, the normal preset is 5 volts. Volts per step is 5.000/4096, or 5000mV/4096, about 1.2207mV, this value was adjusted and used with rounding for 0 volts at counts of 0 and 5 volts at 4095.

Calculate Reading for display as mV or Volts

```
' \ Calculate VstepVar once when Vref is selected
VstepVar = 1.2208 '\ Voltage Step Variable
 ' \ Vvar becomes Voltage in mV
 Vvar = Reading * VstepVal
 Vvar = Int(Vvar + 0.5) '\ Round for 4 digits
 ' \ then display value of Vvar as mV

' \ or change mV to volts
   Vvar = Vvar / 1000
  ' \ then display formatted value as volts
```

Calculate Reading for display as %full scale

```
   Vvar = Reading / 4095
   Vvar = Vvar * 1000
   Vvar = Vvar + 0.5
   Vvar = Int(Vvar)
   Vvar = Vvar / 10
  ' \ then display value as %FS
```

Calculate Reading for display as mA

```
' \ using a 249 ohm 1% resistor
 Vvar = Reading / 204 '\ counts per 1mA
' \ display Vvar as mA of current
```

Setting Analog I/O

An analog output channel is Read/Write. It is set using the Value property.

```
Device1.IoPoints(DA_0).Value = 0
' \ This sets the DA0 Output to 0 counts.
' \ Values from 0 to 1023 are valid. Values are 5V/1024 per step.
Dim Setting as Long
Setting = Device1.IoPoints(DA_0).Value
' \ This returns the current setting of DA0
' \ if want to output mV values from 0 to 5000 mV
DAVoltVar = 5000 ' \ or set another value
```

```
DAVoltVar = DAVoltVar / 4.883
DAVoltVar = Int(DAVoltVar)
Setting = DAVoltVar
Device1.IoPoints(DA_0).Value = Setting
```

A Digital I/O that is set for an Input is Read Only and is read through the Value property for that I/O Point. The result is Boolean.

```
Dim Reading as Boolean
Reading = Device1.IoPoints(DIG_IO_0).Value
```

‘\ This returns True if High and False if Low.

A Digital I/O that is set for an Output is Read/Write and is set/read through the Value property for that I/O Point. The setting/reading is Boolean.

```
Device1.IoPoints(DIG_IO_3).Value = False
```

‘\ Sets Digital I/O 3 Output Low.

```
Device1.IoPoints(DIG_IO_4).Value = True
```

‘\ Sets Digital I/O 4 Output High

```
Reading = Device1.IoPoints(DIG_IO_4).Value
```

‘\ Reading back the Output setting is the same as reading inputs

Reading Multiple Channels

Each channel is consecutive, so all channels can be read into an array.

```
Dim i as Long
Dim Reading(AD_0 to AD_7) as Long
For i = AD_0 to AD_7
   Reading(i) = Device1.IoPoints(i).Value
Next i
```

This creates an array of readings with the same index as the I/O Point ID.

To index readings from 0 to 7, just add the index to the base I/O point AD0 when accessing the I/O Point.

```
Dim i as Long
Dim Reading(7) as Long
For i = 0 to 7
   Reading(i) = Device1.IOPoints(AD_0 + i).Value
```

```
Next I
```

Handling Device Changes – USB Hot Plug/Unplug

The OnChange Event - After `BDaq.BBDaqEnum` is created, any change in the devices connected (plugging in a new device or unplugging a connected device) causes an OnChange event to occur. You can trigger on this event to check available devices, change which devices you are addressing or to detect a new device.

```
Private Sub Connected_OnChange()
```

‘\ Cleanup Variables, Check available Devices,

‘\ Notify the user of the change,

‘\ Change the device being addressed, configure, etc.

```
End Sub
```

If one set of test connections is on the first device installed as “DAQ1” and another set of connections is on “DAQ2” and one is unplugged, we want to set the software to access only the device remaining.

We can use the OnChange event to trigger checking what devices are connected, and which tests to perform.

EnableTest1 is a subroutine that configures the I/O for “DAQ1” and enables the user to start Test 1.

EnableTest2 is a subroutine that configures the I/O for “DAQ2” and enables the user to start Test 2.

```
Private Sub Connected_OnChange()
   Dim Device as Variant
   Set Device1 = Nothing
   Set Device2 = Nothing
   If Connected.Devices.Count = 0 then
    ‘\ Inform user and and stop acess until another connected
      Exit Sub
  End If
  ‘\ If at least one device, check which it is – select test
   For Each Device In Connected.Devices
      If Device.Name = "DAQ1" Then
         Set Device1 = Device
```

```
         Device1.Open
         Call EnableTest1
      End If
      If Device.Name = "DAQ2" Then
         Set Device2 = Device
         Device2.Open
         Call EnableTest2
      End If
   Next Device
   End Sub
```

This routine uses the Count property. The number of connected devices can be found by checking the BBDaqEnum.Devices.Count. Note that disconnecting a device automatically closes that device and accessing any property or calling any function results in an error.

Device Numbers and Names

```
Set Device1 = Connected.Devices(1)
or
Set Device2 = Connected.Devices(2)
or
Set Device2 = Connected.Devices(1)
```

' \ Or By Name: - The named device must be attached

```
Set Device1 = Connected.Devices("DAQ1")
or
Set Device1 = Connected.Devices("DAQ2")
or
Set Device3 = Connected.Devices("DAQ3")
```

' \ 8. If we have one, Open it.

```
   Device1.Open
   Device3.Open
```

Alias Variable Creation for Each I/O Point

Rather than addressing each I/O Point through the Device.IOPoints(ID) method, it is possible to set up a variable for each I/O point. This shortens the time that it takes Visual Basic to access the I/O point.

```
Dim Ain0 As BDaq.BBIoPoint
Set Ain0 = Device1.IOPoints(AD_0)
```

Now any property or method that used to be addressed by Device1.IOPoints(AD_0) can be directly addressed by ain0.

```
Dim Reading as Long
Reading = Ain0.Value
```

This would produce the same result as:

```
Dim Reading as Long
Reading = Device1.IOPoints(AD_0).Value
```

Hardware Timed Reading of Analog Inputs

Any number of Analog Input I/O Points can be read based on a single sampling period. To configure a Device to do timed sampling:

```
' \ 1. Set each Analog Input I/O Point for Single-Ended or Differential as needed through it's Configuration property.
Device1.IOPoints(AD_0).Configuration = AD_12BIT_SE
' \ 2. Add the I/O Point to the list of sampled I/O Points by setting its SamplingEnabled property to True.
Device1.IOPoints(AD_0).SamplingEnabled = True
' \ 3. Set the time between samples by setting the Device's SamplingInterval property. The SamplingInterval property is in seconds.
Device1.SamplingInterval = 0.006
 ' \ This sets sampling all enabled channels every 6 milliseconds.
' \ 4. Start the sampling by setting the Device's SamplingEnabled property to True. Stop sampling when needed by setting to False
Device1.SamplingEnabled = True
Device1.SamplingEnabled = False
' \ 5. Wait for the Device's OnDataAvailable event.
Private Sub Device1_OnDataAvailable()
    '\ get the data here
End Sub
```

When the OnDataAvailable event occurs, get the data. Do this using the GetSamples method for the Device.

The GetSamples method requires one parameter, Count, that specifies the maximum number of samples to return. The Get Samples method returns data for all I/O points being sampled in a two dimensional array. The index for the first dimension of the array is the I/O point. The index for the second dimension of the array is the sample number.

```
Dim SampleReadings() as Long
SampleReadings = Device1.GetSamples(5)
```

Now SampleReadings is an array that contains the next readings after the previous GetSamples method. It will contain readings from the I/O Points on Device1 that have their SamplingEnabled Property set to True.

Note that the array may contain less than 5 samples if the device has not completed 5 sampling periods since the last GetSamples method call. The SampleRedings array is of the form SampleReadings(I/O Point, Sample Number).

For example, if you enable sampling on AD0, AD2, AD3, and AD7 and request 5 samples at a time, then call the GetSamples method, the returned array will look like this

Fig. 5.1 Timed Sample Array

(ch, sam)	Samples Across				
Ch Down	#1	#2	#3	#4	#5
AD0 #1	V1,1	V1,2	V1,3	V1,4	V1,5
AD2 #2	V2,1	V2,2	V2,3	V2,4	V2,5
AD3 #3	V3,1	V3,2	V3,3	V3,4	V3,5
AD7 #4	V4,1	V4,2	V4,3	V4,4	V4,5

If there is an error during the one or more of the GetSamples values, the value is flagged with the `BDAQ_INVALID_DATA_VALUE` instead a value within the normal range below 4096. The most common error is that it can't get the data from the driver and hardware fast enough.

Your program needs to handle indicate where data is missing or invalid and handle the holes.

For example, to display all the readings from AD7 in a text box called txtAD7Result:

```
Dim i as Long
Dim SampleReadings() as Long
SampleReadings = Device1.GetSamples(100)
txtAD7Result = ""
```

```
For i = LBound(SampleReadings, 2) to _
UBound(SampleReadings, 2)
```

‘\ Note: The space underscore (_) is line continuation char

‘\ 2 is the 2nd array element for Lbound and Ubound

```
   txtAD7Result = txtAD7Result & _
   Str(SampleReadings(4, i)) & VBCRLF
```

‘\ without error checking

```
Next I
For i = LBound(SampleReadings, 2) to _
UBound(SampleReadings, 2)
‘ \ obtain value for checking
TempVal = (SampleReadings(4, I)

If TempVal < 4096 Then
   ‘ \ less than error value, use it
   txtAD7Result = txtAD7Result & _
   Str(TempVal) & VBCRLF
Else
 ‘ \ replace with error “****” string
     txtAD7Result = txtAD7Result & _
   “****” & VBCRLF
Endif
```

To display the readings from all enabled channels in the first sample interval since the last GetSample in a text box called txtResult:

```
Dim i as Long
Dim SampleReadings() as Long
SampleReadings = Device1.GetSamples(100)
txtResult = “”
For i = LBound(SampleReadings, 1) to _
UBound(SampleReadings, 1)
   ‘ \ add your error checking
   txtResult = txtResult & _
  Str(SampleReadings(i, 1)) & VBCRLF
Next I
---------------
```

```
When using the GetSamples Method, add the
following to the constant to indicate bad data.

Public Const BDAQ_INVALID_DATA_VALUE As Long =
&H80000000
```

Be sure to check for this invalid value in the examples of timed-triggered sampling.

For a summary of Properties/Methods for the UD modules, look on the CD-ROM disc. We will publish a summary if needed for future models.

This ends the information on the current Library Functions.

Note: During programming, if your program has an error while running within VB, the device may be left open. Then when you fix your error and restart, your program will not be able to Open the device since it was left open. Save your work. Unplug the USB cable, then reconnect it. If you can't recover this way, save your work, exit VB, then restart, re-load your code. It should work normally again.

If your compiled program makes an error and leaves a device open, so it can't be accessed, unplug the USB cable, reconnect.

Only one program at a time can access a UD128A8D device collection.

Chapter 6: *TROUBLESHOOTING*

If you are unable to communicate with the module using your software, please check the results using the BB-DAC software.

As a rule, we are not able to provide programming assistance beyond what is provided in our documentation and example code.

1. Check our Web Site for available FAQ's or troubleshooting hints.
2. Contact B&B Electronics Technical Support for assistance by phone or email.

International Office

Technicians are available at (815) 433-5100 to answer your questions from 8 AM - 5 PM weekdays, Central Time (GMT+5).

European Office

Technicians are available at +353 91-792444 to answer your questions from 8:30 AM – 5 PM weekdays (GMT Time).

Appendix A: *UD128A8D BLOCK DIAGRAM*

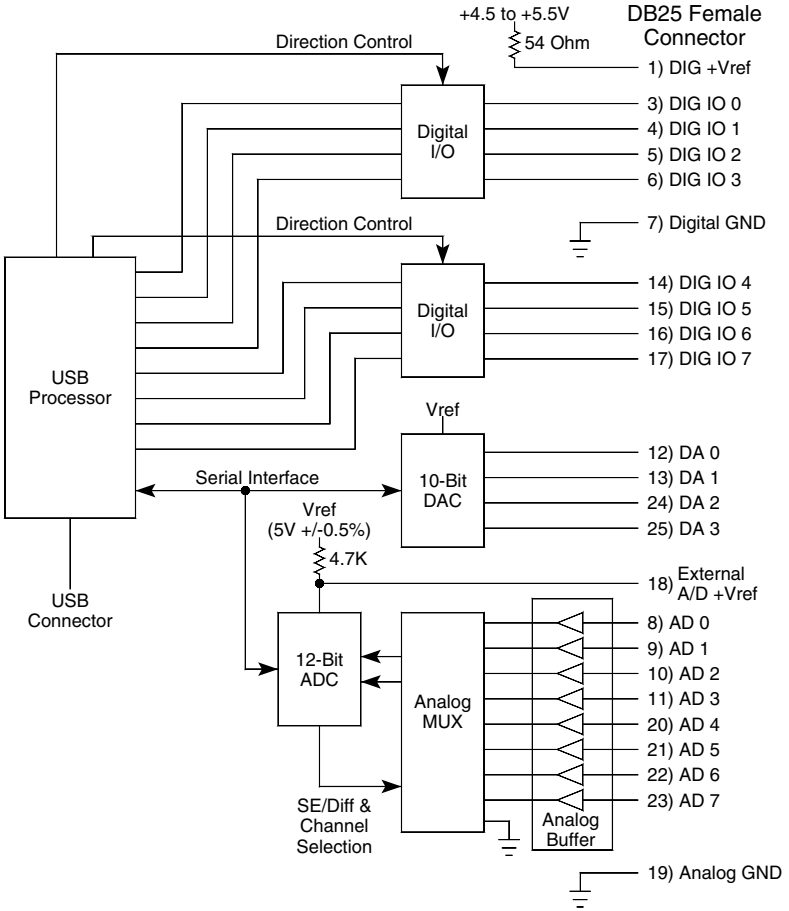


Fig. 1.1 - UD128A8D Block Diagram

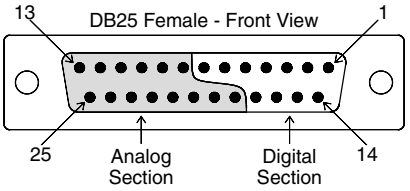


Fig. 1.2 – DB25F I/O Connector Pin Orientation

B&B Electronics Mfg Co – 707 Dayton Rd - PO Box 1040 - Ottawa IL 61350 - Ph 815-433-5100 - Fax 815-433-5104
B&B Electronics Ltd – Westlink Comm. Pk. – Oranmore, Galway, Ireland – Ph 353-91-792444 – Fax 353-91-792445

Appendix B: *I/O PINOUTS*

Description	Direction	Name	DB25F Pin #
+5V Pull-up Digital Vref	Output	DIG +Vref	1
Reserved	----	----	2
Digital I/O Ch0 – Group1	In/Out	DIG_IO_0	3
Digital I/O Ch1 – Group1	In/Out	DIG_IO_1	4
Digital I/O Ch2 – Group1	In/Out	DIG_IO_2	5
Digital I/O Ch3 – Group1	In/Out	DIG_IO_3	6
Digital Ground	---	DIG -GND	7
AD Input Ch0	Input	AD_0	8
AD Input Ch1	Input	AD_1	9
AD Input Ch2	Input	AD_2	10
AD Input Ch3	Input	AD_3	11
DA Output Ch0	Output	DA_0	12
DA Output Ch1	Output	DA_1	13
Digital I/O Ch4 – Group2	In/Out	DIG_IO_4	14
Digital I/O Ch5 – Group2	In/Out	DIG_IO_5	15
Digital I/O Ch6 – Group2	In/Out	DIG_IO_6	16
Digital I/O Ch7 – Group2	In/Out	DIG_IO_7	17
External AD +Vref	Output	ExtAD+Vref	18
Analog Ground	---	A-GND	19
AD Input Ch4	Input	AD_4	20
AD Input Ch5	Input	AD_5	21
AD Input Ch6	Input	AD_6	22
AD Input Ch7	Input	AD_7	23
DA Out Ch2	Output	DA_2	24
DA Out Ch3	Output	DA_3	25

Note: Digital I/O Group #1 or #2 is selectable for inputs or outputs.

Setting For Optional SDDRB4 4-Relay Buffer Board

UD128A8D Signal Name	DB25 Pin#	JP1:Position 12-15	Relay Channel#
DIG_IO_4	14	15	D
DIG_IO_5	15	14	C
DIG_IO_6	16	13	B
DIG_IO_7	17	12	A

Note: Only one SDDRB4 can be set to match outputs

Appendix C: *Declaration of Conformity Statement*

DECLARATION OF CONFORMITY

Manufacturer's Name:	B&B Electronics Manufacturing Company
Manufacturer's Address:	P.O. Box 1040
	707 Dayton Road
	Ottawa, IL 61350 USA
Model Number:	UD128A8D
Description:	USB Data Acquisition System
Type:	Light industrial ITE equipment
Application of Council Directive:	89/336/EEC
Standards:	EN 50082-1
	EN 61000 (-4-2, -4-3, -4-4, -4-6)

CE

William H. Franklin III, Director of Engineering